Notes for Lecture 12

1 Pairings on Elliptic Curves

Let $E/\mathbb{F}$ be an elliptic curve over the field $\mathbb{F}$, and write $E[n]$ for the group of n-torsion points in $E(\overline{\mathbb{F}})$. We recall the fact that if $\operatorname{char} F \nmid n$, $E[n] \equiv \mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/n\mathbb{Z}$. (This "two-dimensionality" of the group of torsion points is what makes interesting pairings possible - there aren't any interesting pairings on cyclic groups!)

Today we'll define the *Weil pairing*

$$e_n : E[n] \times E[n] \to \mu_n,$$

where μ_n is the (multiplicative) group of n^{th} roots of unity in $\overline{\mathbb{F}}$. The Weil pairing satisfies the following properties:

1. e_n is symmetric under inversion: $e_n(P,Q) = e_n(Q,P)^{-1}$;
2. e_n is bilinear (multiplicatively): $e_n(P_1 + P_2, Q) = e_n(P_1, Q) \cdot e_n(P_2, Q)$;
3. e_n is non-degenerate: that is, for any point P of order n there is some point Q such that $e_n(P,Q) \neq 1$.

Note that the addition in the bilinearity property is actually what we'd usually think of as multiplication in μ_n, and the 0 in non-degeneracy is just 1. Recall also that for P to be of order n means that $nP = 0$ but $mP \neq 0$ for all $1 \leq m < n$. So any order n point is n-torsion, but there are n-torsion points that are not order n.

1.1 Divisors

For a point $P \in E(\overline{\mathbb{F}})$, let $[P]$ be a formal symbol. A *divisor* D is a finite integer linear combination of symbols $[P]$, where $\alpha[P] + \beta[P] = (\alpha + \beta)[P]$.

Given a divisor D, we're interested in a few quantities:

- an integer, the degree $\deg D = \sum_{P \in E} \alpha_P$ - note that this is well-defined by the finiteness assumption;

- a finite set of points in $E(\overline{\mathbb{F}})$, the support $\operatorname{supp} D = \{P \in E : \alpha_P \neq 0\}$;
- a point in the elliptic curve, the sum $\Sigma(D) = \sum_{P \in E} \alpha_P \cdot P$, where we use the addition operation in the elliptic curve to add points.

Given two divisors, we can add them as formal linear combinations of points. Divisors form a free abelian group, with identity the divisor with empty support. This group has several important subgroups, all of which are also free abelian.

- the divisors with $\deg D = 0$;
- the divisors with $\Sigma(D) = [O]$;
- the divisors with both $\deg D = 0$ and $\Sigma(D) = [O]$, called the *principal divisors*.

To construct some examples of interesting divisors, let $P \in E[n]$ be an arbitrary nonzero n-torsion point. Write $A = [P] - [0]$, $B = n[P]$, and $C = n[P] - n[0]$. Then $\deg A = 0$ while $\Sigma(A) \neq 0$, and $\deg B \neq 0$ but $\Sigma(B) = 0$. Both $\deg C = 0$ and $\Sigma(C) = 0$, so C is principal.

1.2 Rational functions and divisors

For this section it's probably most useful to just think about $E(\mathbb{C})$, as a lot of the geometric intuition will be lost if you think about $E(\overline{\mathbb{F}})$.

By a *rational function* we just mean a ratio of polynomials

$$f(x, y) = \frac{a(x, y)}{b(x, y)}.$$

We write $f(p)$ for f applied to $p \in E(\overline{\mathbb{F}})$. We call p a *zero* of $f(x, y)$ if $f(p) = 0$, and a *pole* of $f(x, y)$ if $f(p) = \infty$.

Zeroes and poles come with *multiplicities*, and sometimes occur in unexpected places. For example, every polynomial $p(x, y)$ is a rational function - take $b(x, y) = 1$. $p(x, y)$ has a number of isolated zeroes, sometimes with multiplicity. (One possible point of confusion here: ordinarily the solution set to $p(x, y) = 0$ will look like a bunch of curves. But we have to remember we also have a polynomial relation between x and y from the definition of our elliptic curve $E(\overline{\mathbb{F}})$.) More surprisingly, every polynomial has a pole with multiplicity equal to its degree at the point at infinity. (Multiplicities are sometimes called *orders* or *orders of vanishing*.)

To every rational function $f(x, y)$ on an elliptic curve $E(\overline{\mathbb{F}})$ we can associate a divisor

$$\operatorname{div}(f) = \sum_{p \in E(\overline{\mathbb{F}})} \alpha_P [P],$$

where we write

$$\alpha_P = \begin{cases} 0 & \text{if } f(P) \neq 0; \\ +m & \text{if } f \text{ has a zero of multiplicity } m \text{ at } P; \\ -m & \text{if } f \text{ has a pole of multiplicity } m \text{ at } P. \end{cases}$$

Note that $\operatorname{div}(fg) = \operatorname{div}(f) + \operatorname{div}(g)$, and $\operatorname{div}(f) = 0$ if and only if f is constant - this is a nontrivial fact in complex analysis. It turns out (via some more complex analysis) that $\operatorname{div}(f)$ is principal for any rational function f and any principal divisor is $\operatorname{div}(f)$ for some rational function f.

We can now upgrade function evaluation to accept divisors, not points: just define $f(D) = \prod_{P \in \operatorname{supp} D} f(P)^{\alpha_P}$, where we ask that $\operatorname{supp} D$ and $\operatorname{supp} \operatorname{div}(f)$ are disjoint so that this makes sense. Now function evaluation is a homomorphism on the group of divisors: we have $f(D_1 + D_2) = f(D_1)f(D_2)$, though to be completely correct we need to restrict ourselves to the subgroup of divisors whose support is disjoint from that of f.

There's one unusual property of this evaluation map, called *Weil reciprocity*. Let f, g be rational functions, with principal divisors $\operatorname{div}(f), \operatorname{div}(g)$ with disjoint support. Then

$$f(\operatorname{div}(g)) = g(\operatorname{div}(f)).$$

We won't prove this - it turns out to be easy to prove for genus zero curves, but elliptic curves have genus one and a fair bit more work is needed.

2 The Weil Pairing

We'll give two definitions of the Weil pairing: first a naïve definition which will have some obvious flaws, and then the slightly more involved correct definition.

2.1 A naïve definition

For $P \in E[n]$, write $D_P = [P] - [O]$ and $D'_P = nD_p = n[P] - n[O]$. Since P is n-torsion, D'_P is principal, and hence there exists a function f_P with $\operatorname{div}(f_P) = D'_P$ - note the apostrophe on the divisor.

Given points P, Q, define the *naive Weil pairing*

$$e(P, Q) = \frac{f_P(D_Q)}{f_Q(D_P)}.$$

This definition is clearly wrong because f_P has a (n-fold) pole at $[O]$ and D_Q has $[O]$ in its support, but we'll ignore the problem for now and fix it later.

Note that the choice of f_P, f_Q doesn't matter. Indeed, given rational functions f, f' with $\operatorname{div}(f) = \operatorname{div}(f')$, we have $\operatorname{div}(f/f') = 0$, so that f/f' is some nonzero constant function c. Let $D = \sum_{p \in E} \alpha_p [p]$ be a principal divisor with support away from the poles of f (and f'); then

$$(cf)(D) = \prod_{p \in E} (cf(p))^{\alpha_p} = f(D) c^{\sum_{p \in E} \alpha_p} = f(D) c^{\deg D}.$$

But since D is principal its degree is zero, and hence $(cf)(D) = f(D)$. In our context, D'_P and D'_Q are principal, and hence the choice of f_P, f_Q is irrelevant.

It is immediate from our definition of the Weil pairing that it is symmetric up to inversion, namely that $e(P,Q) = e(Q,P)^{-1}$. It's also straightforward to see that $e(P,Q)$ is an n^{th} root of unity whenever P, Q are n-torsion. We have

$$e(P,Q)^n = \frac{f_P(D_Q)^n}{f_Q(D_P)^n} = \frac{f_P(nD_Q)}{f_Q(nD_P)} = \frac{f_P(\operatorname{div}(f_Q))}{f_Q(\operatorname{div}(f_P))} = 1,$$

where the last step uses Weil reciprocity.

2.2 Bilinearity of the Weil pairing

Recall that we want the Weil pairing to satisfy $e(P_1 + P_2, Q) = e(P_1, Q) \cdot e(P_2, Q)$, or equivalently $e(P_1, Q) \cdot e(P_2, Q) \cdot e(Q, P_1 + P_2) = 1$. If we can show this, the Weil pairing will also be linear (multiplicatively) in Q by symmetry up to inversion.

We calculate

$$e(P_1, Q) \cdot e(P_2, Q) \cdot e(Q, P_1 + P_2) = \frac{f_{P_1}(D_Q)}{f_Q(D_{P_1})} \cdot \frac{f_{P_2}(D_Q)}{f_Q(D_{P_2})} \cdot \frac{f_Q(D_{P_1+P_2})}{f_{P_1+P_2}(D_Q)}.$$

Now we introduce the divisor $\Delta = D_{P_1+P_2} - D_{P_1} - D_{P_2} = [P_1 + P_2] - [P_1] - [P_2] + [O]$. Note that Δ is a principal divisor, and so there is some rational function g with $\operatorname{div}(g) = \Delta$. In fact

$$g^n = \frac{f_{P_1+P_2}}{f_{P_1} \cdot f_{P_2}}$$

since $\operatorname{div}(fg) = \operatorname{div}(f) + \operatorname{div}(g)$ and $n\Delta = D'_{P_1+P_2} - D'_{P_1} - D'_{P_2}$.

Then we calculate

$$\begin{aligned} \frac{f_{P_1}(D_Q)}{f_Q(D_{P_1})} \cdot \frac{f_{P_2}(D_Q)}{f_Q(D_{P_2})} \cdot \frac{f_Q(D_{P_1+P_2})}{f_{P_1+P_2}(D_Q)} &= \frac{f_Q(D_{P_1+P_2} - D_{P_1} - D_{P_2})}{g(D_Q)^n} \\ &= \frac{f_Q(\Delta)}{g(nD_Q)} = \frac{f_Q(\operatorname{div}(g))}{g(\operatorname{div}(f_Q))} = 1, \end{aligned}$$

where the last equality uses Weil reciprocity again.

2.3 Fixing the Weil pairing

We return to the problem we noted earlier with the naïve definition of the Weil pairing. Recall that if we are evaluating a function f on a divisor D, we need that none of the poles or zeros of f coincide with any of the points in D. But when we write $f_P(D_Q)$, our function f_P has zeroes and poles at P and O, while D_Q is generated by the points Q and O. So there's a problem at the origin.

We fix this by choosing random points $R, S \in E(\overline{\mathbb{F}})$, and writing

$$D''_P = [P + R] - [R], \quad D''_Q = [Q + S] - [S].$$

Then we define the Weil pairing via

$$e(P, Q) = \frac{f_P(D''_Q)}{f_Q(D''_P)}.$$

This is in fact well-defined, since we've offset the supports of D''_P and D''_Q away from the origin, where f_P and f_Q have poles.

We need to re-check various properties of the Weil pairing, but the calculations are not substantially different than for the naïve definition.

3 Computations via Miller's Algorithm

We've seen that the Weil pairing has the properties that we asked for at the outset. But this is not useful yet, since we don't have a way of computing the Weil pairing. In particular our definition of the Weil pairing asks for a function f_P with $\operatorname{div}(f_P) = n[P] - n[O]$, but we don't have a way of computing such a function.

Miller's algorithm gives a recursive method to compute such an f_P. More precisely, Miller's algorithm computes a sequence of functions $f_{r,P}$ such that

$$\operatorname{div}(f_{r,P}) = r[P] - [rP] - (r - 1)[O].$$

Note that if P is n-torsion, then $\operatorname{div}(f_{n,P}) = n[P] - n[O]$ as desired. We also remark that every $\operatorname{div}(f_{r,P})$ is principal, so our claim makes sense.

We can take $f_{0,P} = 1$, since we only ask $\operatorname{div}(f_{0,P}) = 0$.

To inductively construct $f_{r+1,P}$, we consider the difference

$$\begin{aligned}\operatorname{div}(f_{r+1,P}) - \operatorname{div}(f_{r,P}) =& ((r + 1)[P] - [(r + 1)P] - r[O]) \\ & - (r[P] - [rP] - (r - 1)[O]) \\ =& [P] + [rP] - [(r + 1)P] - [O].\end{aligned}$$

This divisor is principal, so there's some function with zeros and poles at the prescribed points. To find it, we use a bit of cleverness - we find the intersection of the line through the points P, rP and the line through the points $(r+1)P, O$. This intersection is $-(r+1)P$. Then we add and subtract this point to our divisor so that we can split it up as the sum of two (principal) divisors corresponding to lines as follows:

$$\begin{aligned}
&[P] + [rP] - [(r+1)P] - [O] \\
&= [P] + [rP] + [-(r+1)P] - [-(r+1)P] - [(r+1)P] - [O] \\
&= \underbrace{([P] + [rP] + [-(r+1)P] - 3[O])}_{U_{P,rP}} \\
&\quad - \underbrace{([-(r+1)P] + [(r+1)P] - 2[O])}_{V_{(r+1)P}}
\end{aligned}$$

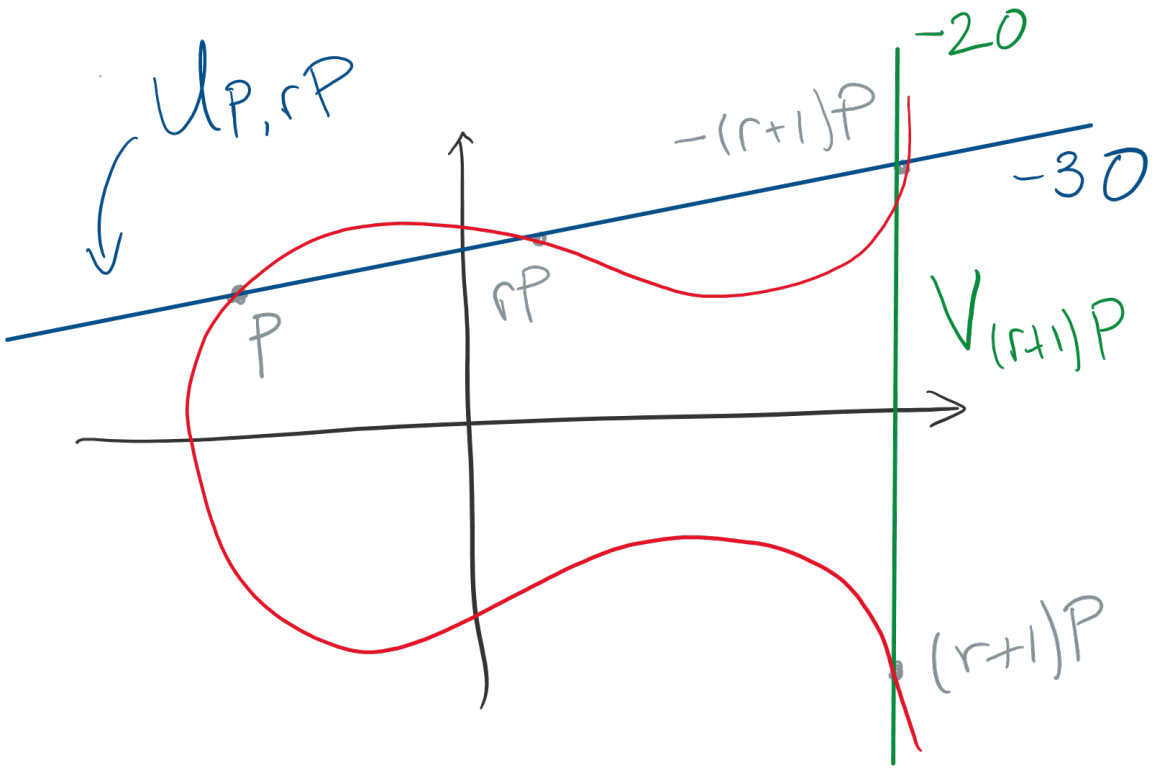

Figure 1: The divisors $U_{P,rP}$ and $V_{(r+1)P}$ on the real points of an elliptic curve $E(\mathbb{C})$.

We can just write down the rational functions in x and y that give these two divisors. For $U_{P,rP}$, we are given the point P, and we can inductively store rP, so it's very easy to write down the equation of a line passing through these two points. For $V_{(r+1)P}$ it's even easier - we just compute $(r+1)P$, and then we get the function $x - x_{(r+1)P}$ for the divisor $V_{(r+1)P}$.

3.1 Miller's algorithm for cryptography

While Miller's algorithm is useful, it's not cryptographically effective in this form, as we generally want n to be very large (at least 2^{128}, possibly on the order of 2^{256}). It takes $O(n)$ calculations to run Miller's algorithm to find a suitable rational function representing a divisor. Furthermore, we've got so many product factors that it takes $O(n)$ time to evaluate our rational function at any given point.

Thankfully, there is a version of Miller's algorithm that computes $f_{2r,P}$ given $f_{r,P}$, and thus makes cryptographic-scale calculations feasible since we only require $O(\log n)$ calculations to find $f_{n,P}$ or evaluate it.

We claim that the function $f_{2r,P}$ defined by

$$f_{2r,P} = f_{r,P}^2 \cdot \frac{U_{rP,rP}}{V_{2rP}}$$

has the required divisor for Miller's algorithm. Indeed, we can check

$$\begin{aligned}
\operatorname{div}\left(f_{r,P}^2 \cdot \frac{U_{rP,rP}}{V_{2rP}}\right) &= 2\operatorname{div}(f_{r,P}) + \operatorname{div}(U_{rP,rP}) - \operatorname{div}(V_{2rP}) \\
&= 2r[P] - 2[rP] - (2r-2)[O] \\
&\quad + 2[rP] + [-2rP] - 3[O] \\
&\quad - [2rP] - [-2rP] + 2[O] \\
&= 2r[P] - [2rP] - (2r-1)[O].
\end{aligned}$$

Here we just require that U is the line through rP and rP; that is, the tangent line to $E(\mathbb{R})$ at rP. The slope of this line can easily be found explicitly using implicit differentiation, so we can write down a rational function representing $U_{rP,rP}$. This lets us calculate the coordinates of the point $2rP$ as well, so we can find a function representing the divisor V_{2rP}.